

REMARKS

Following the filing of an Appeal Brief by applicants, the Examiner reopened the prosecution of the present application by promulgating a final rejection. MPEP 1207.04 permits the Examiner, following the filing of an Appeal Brief, to re-open prosecution with a final rejection only when there is an amendment to the claims or there is newly cited art in an IDS.

In the present application, there have been no amendments to the claims since well before the previous final rejection, and there has been no IDS that cited new art to the Examiner. Accordingly, the only occurrence that could have caused to the Examiner to apply the newly cited Apfel reference is argument made by applicants in their Appeal Brief. The arguments in an Appeal Brief are not by themselves cause to reopen prosecution with a final rejection. Therefore, applicants have given the Examiner no reason to make this present Office Action a final Office Action.

Consequently, applicants respectfully request the Examiner to withdraw the finality of the Office Action, to enter the present amendments, and to consider applicants' arguments and amendments as made herein.

In section 2 of the Office Action, the Examiner objected to claim 1. Claim 1 has been amended to overcome the objection.

In sections 6 and 7 of the Office Action, the Examiner rejected claims 5, 27, and 35 under 35 U.S.C. §112, first paragraph as failing to comply with the written description requirement. These claims have been canceled.

In sections 8-25 of the Office Action, the Examiner rejected claims 1-9, 18, 19, 21, 26-35, and 40-45 under 35 U.S.C. §102(b) as being anticipated by Apfel.

Apfel describes using the Internet to update software or to install new software on a computer 20 either automatically or manually based upon user inputs. The update and new software versions of a Web Authoring Components program 37a can be downloaded via the Internet 60 to the computer 20 from a package server 80b (Figure 2). As Apfel states, an end-user need only transmit a Web site's universal resource locator (URL) in order to access that site.

In order to download the new software or the software updates, the computer 20 sends a query 100 (Figure 3) over the Internet to a database server 80a. The database server 80a is a server at a URL specified in

an auto-autoupdate registry key stored on the computer 20. The URL in the auto-autoupdate registry key may be customized by an administrator at the time of deploying the "WORD 8.0" program module. Also, a predetermined software check date is stored in the auto-autoupdate registry key. The query 100 includes sufficient information regarding the computer 20 so that the database server 80a is able to determine if an update is available and, if so, to determine the location of the update.

The database server 80a returns a response 105 over the Internet to the computer 20. If an update is not available, the response 105 indicates that no update is available and includes a check date upon which an update is expected to be available. If there will be no more updates, or if the administrator turns off the update feature, the response 105 notifies the user that the auto-autoupdate feature is deactivated.

However, if an update is available, the response 105 includes the URL of the update. When the computer 20 receives the URL of the update, the computer 20 sends a query 110 to the package server 80b at the URL of the update. In response, the package server 80b sends

the update 115 to the computer 20 and the computer 20 installs the update 115.

Figures 4A and 4B illustrate a method 400 for automatically updating software or installing new software. At 406, after the word processor program module 37 is booted and the Web Authoring Components program 37a is initialized, a determination is made as to whether the current date is greater than or equal to the date stored in the registry key by the Web Authoring Components program 37a. If the current date is less than the date stored in the registry key, the method 400 ends at 499.

However, if the current date is greater than or equal to the check date stored in the registry key, the method 400 at 409 displays a dialog box warning the user a dial-up connection to the Internet is necessary to proceed and asking whether the user wants to continue. If the user at 412 decides not to proceed, 30 days is added to the check date at 424 to create a next check date stored in the registry key.

However, if the user decides to proceed, a HTTP query is sent at 415 from the computer 20 over the Internet to the database server 80a. Information about the setup of the computer 20 is contained in the query.

This information is needed to determine which update URL is required by the computer 20 and includes the versions of the software to be updated, the platform that the software is running on, and the language of the software.

After the query is sent, it is determined at 418 whether the query was successful in contacting database server 80a. If the query was unsuccessful, an error message at 421 is displayed to the user and 30 days is added at 424 to the check date stored in the registry key.

However, if the query was successful, it is determined at 427 whether an update is available. The database server 80a uses the information received in the query to determine if an update is available, such as by a database lookup. If an update is not available, the database server 80a at 430 returns a "NO UPDATE" message to the computer 20 that includes an encoded check date. The encoded check date is the new check date that an update of the Web Authoring Components program is expected to be available. Also, at 431, an alert is displayed to the user indicating that an update was not available and, after the user acknowledges the alert, the encoded check date received in the "NO UPDATE" message at 432 is stored in the registry key.

However, if a new update is available as determined at 427, the database server 80a at 433 returns an "UP DATE" message to the computer 20. At 439, an update message is displayed to the user asking the user if the user wishes to proceed with the update. The user select either to update the Web Authoring Components program or to terminate the update attempt.

If the user elects not to proceed with the update, 30 days is added to the next check date stored in the registry key (445). If the user elects to proceed with the update, a HTTP download query is sent by computer 20 at 448 to begin downloading the update. At 451, the package server 80b returns the update over the Internet to the computer 20. At 454, the update is installed on the computer 20 and a new check date is stored in the registry key.

As an alternative, the update may be implemented entirely in the background without a dialog box being displayed to the user. Thus, on a specified date, an HTTP query is automatically be initiated and sent in the background. If the query times out, or fails, or a "NO UPDATE" message is returned, there will be no action, and the user will not even know that the query had been initiated. However, if a new software

version is available, the user will be prompted and may choose to automatically apply the update.

Independent claim 1 is directed to a method in which first program code is executed at the content recipient so as to identify a content provider having posted content of interest to the content recipient, second program code is executed at the content recipient so as to automatically initiate a request for the posted content, third program code is executed at the content recipient so as to receive the posted content at the content recipient in response to execution of the second program code, and fourth program code is executed at the content recipient so as to automatically display notice to the content recipient that the posted content has been received at the content recipient in response to execution of the second and third program code.

The Examiner asserts that notice is automatically provided to the user that the update has been received at the user according to the following three portions of Apfel: column 10, lines 2-25; column 10, lines 48-67; and, column 11, lines 1-10. However, these portions of Apfel do not disclose this automatic notice.

Instead, column 10, lines 2-25 merely state that (i) at 432, an encoded date received in a "NO UPDATE" message is stored in the registry key, (ii) at 433, the database server 80a returns an "UP DATE" message over the Internet to the computer 20, (iii) the update message contains the URL of the update to be downloaded, and (iv) after the "UP DATE" message is returned to the computer 20, an update dialog box displays the update prompt text.

As can be seen, the update has not been downloaded yet and, therefore, none of the notices discussed in this portion of Apfel correspond to the notice recited in independent claim 1. That is, the "NO UPDATE" and "UP DATE" messages do not indicate that the update has been received. Thus, this portion of Apfel does not disclose the notice of independent claim 1.

Column 10, lines 48-67 merely state that (i) if it is determined at 442 that the user did elect to download the update, a HTTP download query is initiated and sent by the computer 20 over the Internet to the package server 80b to begin downloading the update, (ii) the HTTP download query is sent to the URL of the update returned in the "UP DATE" message, and (iii) the package

server 80b downloads the update over the Internet to the computer 20.

In this portion of Apfel, the update is now downloaded. However, this portion of Apfel does not describe a notice that the download is received. To the contrary, Apfel states that the download is simply installed. Thus, this portion of Apfel does not disclose the notice of independent claim 1.

Indeed, the final portion of Apfel that the Examiner cites for the notice of independent claim 1 (i.e., column 11, lines 1-10) merely states that (i) at 454, the update is installed on the computer 20, (ii) the new check date for the next update is stored in the registry key, and (iii) the update is designed to decompress itself and automatically install on the computer 20.

As can be seen, this portion of Apfel does not describe a notice that the download is received and, instead, merely states that the download is simply automatically installed. Thus, this third portion of Apfel does not disclose the notice of independent claim 1.

Because Apfel does not disclose the notice of independent claim 1, Apfel does not anticipate independent claim 1.

Moreover, Apfel does not suggest the automatic display of the notice recited in independent claim 1. Apfel instead states at column 11, lines 1-8 that the installation is automatic, suggesting no need to provide the user with notice of the download.

Accordingly, independent claim 1 is not obvious over Apfel.

Independent claim 18 is directed to a computer readable storage medium that stores program code which, when executed, automatically initiates a request for the download of a content element of a web page posted by a content provider, and receives only the content element in response to the request without receiving the whole web page.

There is no disclosure in Apfel that the update described in Apfel is an element of a web page and that only the content element (i.e., not the whole web page) is received.

In rejecting independent claim 18, the Examiner points to column 2, lines 15-40 of Apfel. This portion of Apfel states that (i) a system and method are provided

to automatically or manually update software or install new software, which is communicated over the Internet, on a user's computer, (ii) a determination is made as to whether the current date is on or after a check date stored on the computer 20, (iii) if the current date is on or after the check date stored in the registry key, a database query 100 is sent from the computer 20 over the Internet to a database server 80a, (iv) the database query 100 is a HTTP query and includes information about the software to be updated, the platform that the software is running on, and the language of the software, (v) in response to the query, the database server 80a determines whether an update is available, (vi) if an update is available, the database server 80a sends over the Internet an update message to the user's computer 20, (vii) this message includes the URL of a package server 80b at which the update may be found, and (viii) in response to the update message, the user's computer 20 sends over the Internet an update query 110 to the package server 80b.

As can be seen, there is no mention here that the update is an element of a web page that is downloaded to the user's computer separately from the web page.

Accordingly, this portion of Apfel does not disclose the invention of independent claim 18.

The Examiner also points to column 5, line 50 through column 6, line 5 of Apfel. This portion of Apfel states that (i) the Internet includes a plurality of Internet sites 75a through 75n, (ii) the Internet sites 75a through 75n are generally operated by corporations, universities, and governmental organizations, (iii) each Internet site may include one or more repositories of information and resources that may be accessed over the Internet, (iv) each Internet site may include a plurality of web servers 80a through 80n, (v) each of these web servers 80a through 80n may provide "home pages" to be visited, files to be read or downloaded, applications to be shared, and the like, and (vi) the Internet also includes a plurality of points of presence 85a through 85n that are operated by local access providers.

As can be seen, there is no mention here that an update is an element of a web page and can be downloaded to the user's computer separately from the web page. Accordingly, this portion of Apfel likewise does not disclose the invention of independent claim 18.

Because Apfel does not disclose that the update is an element of a web page and can be downloaded to the

user's computer separately from the web page, Apfel does not anticipate independent claim 18.

Moreover, Apfel does not suggest that the update is an element of a web page that is downloaded to the user's computer separately from the web page. Apfel instead describes home pages without once even hinting that the update can be an element of a home page.

Accordingly, independent claim 18 is not obvious over Apfel.

Independent claim 32 is directed to a method comprising executing first program code at a content provider so as to post content for access by a content recipient, executing second program code at the content recipient so as to automatically (i) access the content provider and (ii) initiate receipt by the content recipient of the posted content if the posted content is new, executing third program code is executed at the content provider so as to send a message notifying the content recipient that the posted content is not new, and electronically receiving the second program code at the content recipient.

The Examiner asserts that Apfel at column 6, line 63 through column 7, line 10 discloses the feature of electronically receiving the second program code at

the content recipient. This program code is executed at the content recipient and causes the automatic accessing of the content provider and the initiation of the receipt by the content recipient of the posted content if the posted content is new.

Column 6, line 63 through column 7, line 10 state that (i) if an update is available, the database server 80a sends back a response 105 that includes the URL of the update, (ii) the database server 80a also sends back the URL of an alternative download site in case the primary download site is overloaded, (iii) after the computer 20 receives the response 105 including the URL of the update, the computer 20 sends a query 110 to the package server 80b at the URL of the update, (iv) the package server 80b sends the update to the computer 20, and (v) the computer 20 installs the update.

As can be seen, Apfel does not disclose here or anywhere that the program code, which causes the database server 80a to be automatically accessed and the receipt of the update by the computer 20 to be initiated by the computer 20, is electronically downloaded to the computer 20. Moreover, there is no disclosure in Apfel that the update contains this program code.

Because Apfel does not disclose electronically downloading the second program code of independent claim 32 to the computer 20, Apfel does not anticipate independent claim 32.

Moreover, Apfel does not suggest this feature of independent claim 32.

Accordingly, independent claim 32 is not obvious over Apfel.

Independent claim 45 is directed to a method performed at a content recipient comprising (i) executing first program code at the content recipient so as to identify a content provider having posted content of interest to the content recipient, (ii) executing second program code at the content recipient so as to automatically initiate a request for the posted content and to automatically download the posted content if the posted content is new, and (iii) executing third program code at the content recipient so as to receive a notice that the content provider has no new content to download to the content recipient.

Apfel in Figures 4A and 4B discloses that the user at 442 must select either to proceed with the download of the update or to terminate the update process entirely. Accordingly, this selection requires a manual

operation and, therefore, cannot be automatic contrary to the recitations in independent claim 45.

Even in the alternative proposed by Apfel in column 11, lines 49-59, the download is not automatic. Instead, the user is prompted to download the update. (The automatic application mentioned in this portion of Apfel merely relates to the automatic installation once the update is downloaded.) This prompting requires a manual selection on the part of the user such that the downloading of the update cannot be automatic.

Accordingly, independent claim 45 is not anticipated by Apfel.

Moreover, Apfel does not suggest automatic download. To the contrary, even when Apfel suggests transparency to the user, Apfel still teaches that the user should still be offered the selection of whether to download the update.

Accordingly, independent claim 45 is not obvious over Apfel.

Dependent claims 19 and 21 recite the provision of a notice that the posted content has been received in response to the request. As indicated above in connection with independent claim 1, Apfel does not disclose or suggest the provision of a notice that the

posted content has been received in response to the request.

Accordingly, dependent claim 19 is patentable over Apfel.

Dependent claim 30 recites that the stored program code is electronically received from a remote site and is stored by the computer readable storage medium. As indicated above in connection with independent claim 32, Apfel does not disclose or suggest electronically receiving the stored program code from a remote site and storing the electronically received program code on the computer readable storage medium.

Accordingly, dependent claim 30 is patentable over Apfel.

Dependent claim 40 recites that fourth program code is executed at the content provider so as to determine whether the content recipient possesses the second program code and, if the content recipient does not possess the second program code, to download the second program code to the content recipient.

However, as discussed above in connection with claim 32, Apfel does not disclose or suggest that the program code is electronically received by the computer 20. Therefore, Apfel cannot disclose or suggest that the

database server 80a or the package server 80b determines whether the computer 20 possesses the program code that causes the package download to be initiated.

The Examiner asserts that Apfel discloses these features at column 6, line 35 through column 7, line 30. However, the Examiner does not explain how this portion of Apfel relates to the features of dependent claim 40. Indeed, applicants cannot find the features of dependent claim 40 anywhere in this passage.

Accordingly, dependent claim 40 is patentable over Apfel.

Dependent claim 41 recites that the second program code is electronically received at the content recipient from the content provider.

As can be seen from the above discussion in connection with independent claim 32, dependent claim 41 is patentable over Apfel.

Dependent claim 43 recites that, upon an action related to the notice, the posted content is displayed to a user.

To meet dependent claim 43, Apfel would have to disclose that the update is displayed to the user of the computer 20. However, there is no disclosure or

suggestion in Apfel that the update is displayed at all, much less in response to an action related to a notice.

The Examiner asserts that this feature is disclosed in column 7, lines 40-55 and column 11, lines 1-10 of Apfel.

Column 7, lines 40-55 of Apfel state that (i) a determination is made as to whether the current date is greater than or equal to a date stored in a registry key, (ii) the registry key is set equal to the date that an update to the Web Authoring Components program module is expected when the Web Authoring Components program module is initially installed, and (iii) the current date is the date that is maintained on the internal clock of computer 20.

As can be seen, this portion of Apfel does not describe an action, which is related to a notice indicating the download of content, that causes the content to be displayed.

Column 11, lines 1-10 of Apfel state that (i) the update is installed on the computer 20, (ii) a new check date is stored in the registry key, (iii) this new check date may be randomized so that the server is not overloaded, and (iv) the update is designed to decompress itself and automatically install.

As can be seen, this portion of Apfel likewise does not describe an action, which is related to a notice indicating the download of content, that causes the content to be displayed.

Accordingly, dependent claim 43 is patentable over Apfel.

Dependent claim 44 recites that the content element comprises a note attached to the web page.

The Examiner points to column 2, lines 15-60 of Apfel. However, as discussed above, this portion of the Apfel does not disclosed that the update or anything else that is downloaded to the computer 20 is attached to a web page.

Accordingly, dependent claim 44 is patentable over Apfel.

In sections 26-31 of the Office Action, the Examiner rejected claims 12, 13, 15, 16, 20, 22, 24, 25, 28, 36, 37, and 39 under 35 U.S.C. §103(a) as being unpatentable over Apfel in view of Stephens.

However, Stephens does not make up for the deficiencies of Apfel with regard to independent claims 1, 18, 32, and 45. Therefore, independent claims 1, 18, 32, and 45 are patentable over Apfel in view of Stephens. Because independent claims 1, 18, 32, and 45 are

patentable over Apfel in view of Stephens, dependent claims 12, 13, 15, 16, 20, 22, 24, 25, 28, 36, 37, and 39 are *per force* patentable over Apfel in view of Stephens.

In section 32 of the Office Action, the Examiner rejected claims 14 and 23 under 35 U.S.C. §103(a) as being unpatentable over Apfel in view of Stephens and further in view of Beyda.

However, Beyda does not make up for the deficiencies of Apfel and Stephens with regard to independent claims 1, 18, 32, and 45. Therefore, independent claims 1, 18, 32, and 45 are patentable over Apfel in view of Stephens and further in view of Beyda. Because independent claims 1, 18, 32, and 45 are patentable over Apfel in view of Stephens and further in view of Beyda, dependent claims 14 and 23 are *per force* patentable over Apfel in view of Stephens and further in view of Beyda.

CONCLUSION

In view of the above, it is clear that the claims of the present application patentably distinguish over the art applied by the Examiner. Accordingly, allowance of these claims and issuance of the above captioned patent application are respectfully requested.

Respectfully submitted,

SCHIFF HARDIN LLP
6600 Sears Tower
233 South Wacker Drive
Chicago, Illinois 60606
(312) 258-5500
CUSTOMER NO. 32692

By: 

Trevor B. Joike
Registration No.: 25,542
Attorney for Applicants

September 26, 2006